Episode Book

EPISODE #6654

Two Ways AI Edits Audio From Plain Instructions

PUBLISHED	17 November 2026 at 10:05 UTC
DURATION	22:48
LISTEN	myweirdprompts.com/episode/instruction-based-audio-editing-39686/
HOSTS	Corn, Herman, Dorothy, Hilbert
SCRIPT MODEL	z-ai/glm-5.3
VOICES	omnivoice
PIPELINE	V5.35
BOOK GENERATED	17 November 2026 at 14:42 UTC

SYNOPSIS

Instruction-based audio editing promises to change a clip using plain language, but the label hides two completely different approaches. Generative models like Microsoft and CUHK-Shenzhen's AUDIT, StepFun's Step-Audio-EditX, and Tencent's AuK re-synthesise audio, which risks unintended drift in the original take. Meanwhile, parametric systems like Sony AI's LLM2Fx and Northwestern's Text2FX leave the signal untouched and adjust conventional EQ and reverb knobs instead.

TRANSCRIPT



Corn

April twenty twenty-three. A research team publishes a model that takes your audio clip, reads the sentence "remove the dog barking," and hands you back the audio without the dog. Three and a half years later, there is still no product you'd trust with a client's take.



Herman

Which is exactly the gap Daniel's poking at, because his message this week is about that whole category. Instruction-based audio editing. You hand the model a clip plus a plain-language instruction — his examples are "make the voice sound bassier," "cut the low end," "remove the dog barking," "make her sound happier" — and it returns edited audio. And he's asking three things, really: what is this category, what is it actually used for, and how does it differ from the audio-to-audio tasks he already knows — speech enhancement, source separation, voice conversion.



Corn

And the interesting part isn't the demos. It's that "edit" is hiding two completely different machines under one word, and most of the field is research papers rather than shipping products.



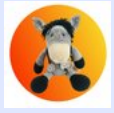
Herman

So let's draw the category properly, because it's narrower than it sounds. The defining constraint is that you supply an existing clip and describe only the change. Not the target — the delta. "Cut the low end" is an instruction. "Here's the clean audio I want" is not. The model has to work out what you mean and apply it to your material.



Corn

You're not prompting a generator from nothing. You're pointing at something that already exists and saying, that, but different.



Herman

And the moment you draw that box, everything inside splits into two families. One: generative editors. These re-synthesise the clip — they regenerate your audio conditioned on the instruction. Two: LLM-driven effect control. These don't regenerate anything. A



Herman (continued)

language model reads your instruction and turns knobs on conventional effects — EQ, reverb, compression — applied to your original signal.



Corn

Two families, one word. Keep that split in your head, because everything today hangs off it.



Herman

The early reference point for the generative side is AUDIT, from Microsoft and CUHK-Shenzhen, back in April twenty twenty-three, published at NeurIPS that year. It's a latent diffusion model — takes the instruction plus the input audio, and its task list is basically the template the whole field still uses: add a sound, drop a sound, replace one sound with another, inpainting, and super-resolution.



Corn

And can you download it?



Herman

No. That's the pattern, actually. It's a demo page. No confirmed public weights. AUDIT is a paper you can read and a set of examples you can listen to, and that's where it stops.



Corn

A museum piece you can hear but not touch.



Herman

Now, the honest state of the field — and this matters for Daniel's question, because he's a Hugging Face person. If you go to the audio-to-audio task listing on Hugging Face today, what's trending is mostly voice conversion, source separation, and speech enhancement. Seed-VC, Demucs, the BS-RoFormer variants. Instruction-following editors are rare on that listing. They mostly live as research releases, or under other tags entirely.



Corn

So the category exists, the demos work, and the shelf is empty.



Herman

Largely. Let's take the generative family seriously first, because that's where the demos live. And it splits again, between music and speech.



Corn

Music first.



Herman

Music editing by instruction has a decent run of papers. InstructME, at IJCAI twenty twenty-four, does instruction-guided music editing on a latent diffusion backbone — add instruments, remove them, extract, replace, remix — and it conditions on chords, which is a nice touch, because "remove the drums but keep the harmony intact" is the kind of thing a musician would actually ask for. Trained on four hundred and seventeen hours of music.



Corn

Four hundred seventeen. Someone labelled all of that.



Herman

And no public code, as far as I've ever been able to find. Then Instruct-MusicGen, arXiv twenty forty-five point eighteen three eight six, ended up at ISMIR twenty twenty-five — that one takes Meta's MusicGen, which everyone knows, and adapts it to add, remove or separate stems by instruction. The clever part of that lineage is it's standing on a model that already knows music.



Corn

The speech side is where it gets more interesting, because that's where the open weights actually are.



Herman

Yes! Step-Audio-EditX from StepFun. The tech report landed the sixth of November last year, weights six days later, code under Apache two point zero. It's an LLM-based speech editor, about three billion parameters, and the scope is what makes it notable — it edits emotion, so angry, happy, sad; speaking style, so whisper, child, older; and the paralinguistic stuff, which is the part I find cool — breathing, laughter, sighs, the "uhm." And it's iterative. You can say "make her happier," listen, then say "happier still" and it applies on top.



Corn

So it's a conversation with the take. Edit, listen, edit again.



Herman

That's the pitch. The model card is honest about the limits, though — clips under thirty seconds, whisper needs several passes, and it wants about twelve gigabytes of GPU memory. And I should say the comparisons it makes against MiniMax and Doubao are the authors' own claims, not independent tests.



Corn

Three billion parameters to say "sadder, but the same sentence."



Herman

There's also Tencent's AuK — one point five billion, MIT licence on the model card — which is more surgical: replace, insert or remove words, change pitch, speed, volume, convert a take to a whisper, and it preserves the rest of the take. And dots.tts.edit, from dots-studio on Hugging Face, does instruction-controlled speech editing in English and Mandarin at forty-eight kilohertz — text edits, emotion, prosody, pauses, enhancement, even background audio.



Corn

Okay, hold on. Before we get deeper — voice conversion. Daniel specifically asked how this differs, and I want to get the boundary right, because on the surface "make her sound happier" and "make her sound like him" look like the same shape of request.



Herman

It's a good boundary to draw. Voice conversion changes who is speaking. It's a fixed identity mapping — one voice in, another voice out, and the model has exactly one job forever. Instruction editing changes properties of the same take — the emotion, the style, whether the dog is in it — and you can iterate. The identity stays put.



Corn

So voice conversion is a costume. Instruction editing is direction.



Herman

I'd sign that. And the direction metaphor holds, because with instruction editing the model has to understand the scene before it can follow the direction. That's the real technical crux, and it's the thing I find most interesting in the whole research line. MMEdit — that's arXiv twenty-five twelve point two zero three three nine, December last year, still under review — pairs a Qwen2-Audio encoder with an MMDiT generator. And their core argument is that a text-only encoder can't find the target event in a busy scene.



Corn

Meaning what, concretely?



Herman

Meaning "remove the dog barking" fails if the model can't locate the dog. You've got a clip with a voice, traffic, a fridge, and a dog. The instruction is about one event in that mixture. If the model's understanding of the clip comes only from text — from a caption of the audio — the caption might mention the dog or might not, and even if it does, "the dog" as a phrase doesn't point at three seconds in the middle of the take. So MMEdit gives the model an actual audio encoder so it can ground the instruction in the signal.



Corn

The model has to answer "where" before it can answer "what."



Herman

Right. And there's a second problem the whole field shares, which is where the training data comes from. You need pairs — input, instruction, output — and nobody has sat down and recorded a "with dog" and "without dog" version of thousands of takes. So the trick everyone uses is synthetic: you apply a known edit and write the instruction for it afterwards.



Corn

Run an effect chain over a clip, and the instruction becomes "cut the low end," because you're the one who cut it.



Herman

That's it. SAO-Instruct — NeurIPS twenty twenty-five, fine-tunes Stable Audio Open — builds its triplets that way, with Prompt-to-Prompt, DDPM inversion, and a manual pipeline. The authors say it generalises to real clips, which is the claim that matters, because a model trained only on its own edits could just be learning to undo them. They say they release code and weights, though I haven't confirmed the licence, and I'd note the base model's Stability Community licence restricts commercial use either way.



Corn

Same argument you and I have had about model-card licences before — check the card, not the badge on the listing.



Herman

Now — the cost of all this, stated plainly, because it's the defining weakness of the family. The generative editor re-synthesises the clip. That's what lets it change any event, any emotion, any voice. But regeneration doesn't stay inside the lines you drew. It can drift in the parts you didn't ask to change. The voice timbre shifts a little, the room changes, a consonant softens. That's not a bug in one model. That's the price of the approach.



Corn

Every edit is a re-recording, whether you asked for one or not.



Herman

And now it's time — Dorothy's History of Toast. Mum, you're on.

D

Dorothy

Hello dear, it's Dorothy with Dorothy's History of Toast. Mrs Beeton's Book of Household Management, from eighteen sixty-one, has a recipe for a toast sandwich.

D

Dorothy

One slice of toast between two slices of bread, and it's recommended for invalids.

D

Dorothy

You'd think toast would be a comfort, but no, it turns out it's mostly for people who are already poorly. Things do not work out, sweetheart.

D

Dorothy

Anyway, I've got the kettle on and your father's socks to pair. Ring me when you can.



Herman

A toast sandwich. Bread as the bun for toast.



Corn

Invalids deserved better.



Herman

Where was I — drift. Yes. So here's the thing about drift: it connects to something we talked about ages ago, on the episode about whether a voice fine-tune becomes useless as your voice ages. Same disease, different patient — the model doing something to



Herman (continued)

audio that the owner of the audio didn't sanction.



Corn

And in both cases the fix is the same unglamorous one: someone has to listen and compare. There's no metric for "he didn't ask for that."



Herman

Okay — the other family. And honestly, Corn, this is the sleeper story, because it's less glamorous and much closer to working. The parametric family doesn't touch the signal at all. A language model reads your instruction and sets conventional effects on your original audio.



Corn

So "make it bassier" becomes... a low shelf, plus two dB.



Herman

Exactly that. LLM2Fx from Sony AI — WASPAA twenty twenty-five — maps a text description straight to EQ and reverb settings, outputs them as JSON, and does it with no fine-tuning. It's all in-context: the model gets DSP feature statistics, DSP code, and a few examples, and reasons its way to the parameters.



Corn

No training run. It just... knows what "warm" means for a filter.



Herman

Because it's read the code! That's the part I love. The effect's behaviour is described to it as code, and the language model — which has spent its life reading code — reasons about what parameter values produce the described result. Then Text2FX, from Northwestern, takes the opposite approach: it uses CLAP audio-text embeddings and differentiable DSP, and optimises the EQ and reverb parameters by gradient descent toward the text



Herman (continued)

description. So instead of the LLM guessing the numbers, the numbers are nudged until the audio lands near where the text sits in embedding space.



Corn

Two routes to the same knob. One reasons it out, one hill-climbs it.



Herman

And InstructFX2FX, a DFx demo from this year, combines them — LLM plans the effect chain, CLAP-guided optimisation refines it, and you get multi-turn editing. Though the authors themselves flag drift and instability with non-differentiable effects. Anything that's a commercial plugin with no gradient, you need gradient-free search — Bayesian optimisation, that's what FxSearcher does.



Corn

Which is the most electronic-music sentence you've said all year, and I want you to know I've watched you build pedalboards with less.



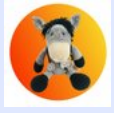
Herman

The pedalboards worked.



Corn

The pedalboard worked. The power supply is still in the fruit bowl.



Herman

It's a good fruit bowl. Anyway — here's the trade-off that defines the whole category, and it's the cleanest way I know to answer Daniel's question about what separates all these things. Generative editors can change any event, any voice, any emotion — but they re-synthesise, so they can drift. Parametric editors keep your original signal and are predictable — but they only do what their effects can do. "Make it bassier" is effectively a solved parametric problem. "Remove the dog barking" needs a generative model.



Corn

One sentence, and the whole map falls out. Bassier is an EQ curve. The dog is a scene.



Herman

And that's why both families exist and why neither is going away. Now the fixed-function comparisons, because Daniel named them specifically and they deserve a proper answer, not a hand-wave. Speech enhancement — denoise, dereverb — has one job and no instruction channel. You don't tell it what you want. You pick the model, and it does the thing it does, forever.



Corn

There's no text box. The "interface" is which model you downloaded.



Herman

Source separation is more capable but still not an edit. It splits a mixture into stems — Demucs v4 and the BS-RoFormer variants are what actually trends — and that's a decomposition. It hands you the voice and the dog and the traffic on separate tracks, and then you decide what to do. It doesn't know what you want changed; it knows everything that's there.



Corn

So separation is the ingredient, and the instruction is the recipe.



Herman

Instruction editing is the only one of the three where the target is specified in language at inference time. Which is precisely why it's the hardest to make reliable — with the fixed-function tools, the engineering team chose the behaviour in advance and froze it. With an instruction editor, the user invents the behaviour at the moment of use, and the model has to cope.



Corn

The fixed tools are appliances. This one's an employee.



Herman

And the honest state of play on the "can you use it" question: most of the named systems are research releases with demos. Open weights are the exception — Step-Audio-EditX, AuK, dots.tts.edit, and SAO-Instruct claims to release them. But no one has shown a general-purpose, production-grade, do-anything instruction editor. The category exists. The demos work. The product doesn't.



Corn

Yet.



Herman

Yet. And here's the second-order thing I keep landing on. The parametric route is cheap and predictable. The generative route is expensive and drifty. So the near-term deployment is almost certainly hybrid — a router, some LLM sitting in front, that sends "make it bassier" to an effect chain and "remove the dog" to a generative model. And the interesting question isn't which family wins.



Corn

It's who writes the router.



Herman

Right, Corn, and here's where I actually want your view, because you're the one in this flat who distrusts anything that edits while you're not watching. If the router mis-routes — if "clean up the room tone" gets sent to the generative model instead of the EQ — does the user ever find out?



Corn

Only if they listen to the whole take. And nobody listens to the whole take. That's the trap. The failure isn't bad audio — bad audio is loud, you notice it. The failure is a confident, quiet, plausible edit in the wrong place, in a take you stopped monitoring forty minutes ago. I'd catch it, because I listen to everything twice at my speed. You'd catch it about Thursday.



Herman

That's fair. I install the thing the week it ships and I've usually broken my setup with it before it comes up on the show. You know my habits better than the router would.



Corn

The router would learn them. That's what worries me.



Hilbert

It's twelve gigabytes. EditX. You said ten earlier — it's twelve on the model card.



Herman

Twelve. Thank you, that's the figure.



Hilbert

I've got a take I've been fixing for a while. One take. There's a fridge in it, and I've been asking one of these models to take the fridge out. It takes the fridge out. It also does something to the room. Small. But it's not the room I recorded.



Herman

The room tone shifts.



Hilbert

Every time. And I've concluded it's because the model doesn't know which sound is the fridge. It's guessing, same as MMEdit said. So I keep a list now — every instruction I've tried, and beside it everything the model changed that I didn't ask for. I don't accept an edit off the first pass. I record a second take of every room, silent, nothing happening, just the room, and I play that back before I accept anything.



Corn

You re-record the room itself as a reference.



Hilbert

It's the only way to know what nothing is supposed to sound like. Herman, you'd have called that gain staging thirty years ago. Same discipline, daft machinery.



Herman

It is the same discipline. Corn's silent-second idea is cheaper, but a reference take of the room is better than measuring it.



Corn

For the record, my silent-second idea was for the kitchen door slam, and it worked.



Hilbert

The list is up to forty-one entries now. Last one, the model took the fridge out and put a faint hiss in where the fridge was. I've written to the people who make it. They said it's within normal behaviour.



Corn

Forty-one. That's a hand-kept drift log for one clip.



Herman

Which is the thing nobody in the research pack has claimed to solve. His list is the gap, itemised.



Corn

And it's the answer to my router worry, really — he's the user who listens, and it still took him forty-one attempts to get one fridge out.



Herman

So the "one thing" — if you take one thing from today: "instruction-based editing" is two machines, not one. The effects-side machine keeps your signal and can't remove a dog; the generative machine removes the dog and re-records everything else around it. The word "edit" hides which one you got.



Corn

The version you got is the one you have to listen all the way through to trust.



Herman

The open question I'm leaving with: if the near-term shape is a router, the interesting failure isn't bad audio — it's a confident routing decision the user never sees. Whether that's a dial you expose or a decision you bury is going to separate the products from the demos, and I don't think anyone's designed that dial yet. Hilbert's forty-one-entry list is what the user does until the dial exists.



Corn

Which also means the gap between a demo and a product is measured in exactly one thing — whether the parts you didn't ask for stay put.



Herman

Producer credit to Hilbert Flumingtop, who has been at the desk and, evidently, at the fridge. If you enjoyed this one, try episode three, Safetensors or something else; episode seven, Building Custom ASR Tools; and episode fifteen, AI Gets Personal. This has been My Weird Prompts — no question too obscure, no rabbit hole too deep. If the show's been worth your commute, a review on Spotify helps other people find us. We'll be back soon.



Corn

See you tomorrow, whether the fridge is still there or not.